Looking Back: Alan Turing–The Father of Computer Science

Article *in* CSI Communications · January 2013

CITATION
1

READS
3,088

1 author:

Vivek Kulkarni
HP Inc.
3 PUBLICATIONS **88** CITATIONS

SEE PROFILE

Article

Vivek Kulkarni
Principal Architect - LABS, Persistent Systems

Looking Back: Alan Turing- The Father of Computer Science

Computers: We use them every dayfrom education to games, offices, homes, banks, movie theaters and more.Our ability to incorporate computers in our everyday lives is largely due to the contributions of one individual, Alan Turning.V

Alan Turing, the founder of Computer Science, a mathematician, philosopher, code-breaker, was an extraordinary visionary of his time. His invention, the Universal Turing Machine laid the foundation for today's digital computers. We would not be where we are today without his contributions in computability theory.

We often take for granted technological advances that make our day to day lives easier. Most times we attribute these contributions to people like Steve Jobs and Steve Wozniak; however, before the invention of the computer, it was necessary for someone needed to introduce the idea of the computer.

On June 23, 1912 Alan Mathison Turing was born in an upper-middle class british family in Paddington, London[1]. In school, Turing's scientific talents were not appreciated by more conservative teachers, leading him to make science ashis extra-curricular passion. Often absorbed in his own ideas, Turing was criticized for his handwriting, struggled with English, and even in mathematics. Despite being an unconventional student, Turing won almost every possible prize for mathematics in addition to being amazing athlete.

Two years after his graduation in 1936, Turing published his most important paper, *On Computable Numbers, with an Application to the Entscheidungsproblem*[2]. *Entscheidungsproblem* is the German word meaning decision problem, a challenge posed by David Hilbert in 1928, a proposition for an algorithm that decides whether a given mathematical statement is provable from the axioms using the rules of logic[3]. In this paper, Turing introduced what later became known as Turing machine, a hypothetical device capable of performing any mathematical function, ultimately laying the foundation for the concept of an algorithm (a set of instructions for the machine to follow). The most striking positive result concerning the capabilities of Turing Machines is the existence of Universal Turing Machines (UTM)[4]. The UTM is the machine capable of simulating any other Turing Machine; in essence a programmable computer. When a UTM is given a program (a description of another machine), it makes itself behave as if it is that machine while processing the input.

Alonzo Church, whose work on Lambda Calculus intertwined with Turing's work in a formal theory of computation, also accepted Turing's claim in what is known as the *Church's Turing Thesis*. The basic idea is that Turing Machines capture the informal notion of effective method in logic and mathematics, and provide a precise definition of an algorithm or 'mechanical procedure'. Turing also put forth the concept of an unsolvable problem[5]. According to Turing,it is impossible to decide whether a Turing machine with a given table of instructions will output an infinite sequence of numbers[6]. This problem is referred as the *Halting Problem* and is incomputable. Subsequently, the concept of the Turing machine has become the foundation of the modern Theory of Computation, a theory central to Computer Science.In principle, Turing created the theoretical framework for computers in his paper, much before the first digital computer was built.

Turing's 1945 report entitled "Proposed Electronic Calculator" gave the first relatively complete specification of an electronic stored-program general-purpose digital computer[7]. His idea of a machine was called the *Automatic Computing Engine (ACE)*.In the U.S., the first electronic stored-program digital computer was proposed by von Neumann in his report titled "First Draft of a Report on the EDVAC" in May 1945[8]. EDVAC stands for *Electronic Discrete Variable Automatic Computer*. This paper achieved greater circulation and the computer architecture it outlined became known as the "von Neumann architecture" though it contained a little engineering detail.Many people have acclaimed von Neumann as the "father of the computer" but he was well aware of the fundamental importance of Turing's paper of 1936.

During the Second World War, Turingworked at Bletchley Park. It was the Britain's code-breaking centerwhere the brightest minds in the country including Gordon Welchman and Harold Keen collaborating to crack German ciphers. Together they invented the Bombe - a machine designed to break Enigma, the

1. Turing's Biography -http://www-groups.dcs.st-and.ac.uk/history/Biographies/Turing.html
2. http://en.wikipedia.org/wiki/Entscheidungsproblem
3. Formal system -http://en.wikipedia.org/wiki/Logical_system
4. UTM is the machine capable of simulating any other Turing Machine. That means, it can simulate any algorithm possible.
5. Unsolvable problem is the one for which no algorithmic solution exists.
6. Halting problem - http://en.wikipedia.org/wiki/Halting_Problem
7. Proposed Electronic Calculator - http://www.alanturing.net/turing_archive/archive/p/p01/p01.php
8. http://www.fh-jena.de/~kleine/history/machines/Godfrey-VonNeumannComputerPlan.pdf

German's primary method of encryption. It is believed that these efforts shortened the warby at least two years[9].

Turingbegan to consider the issue of artificial intelligence, formulating the famous Turing Test[10]. Introduced by Alan Turing in his 1950 paper, *Computing Machinery and Intelligence*, The Turing Test is anassessment of a machine's ability to exhibit intelligent behavior which opens with the words: "I propose to consider the question, 'Can machines think?'"[11] Since "thinking" is difficult to define, Turing chooses to replace the question by another, which is closely related to it and is expressed in relatively unambiguous words. Turing's new question was: "Are there imaginable digital computers which would do well in the imitation game?"In the years since 1950, the test has proven to be both highly influential and widely criticized, and it is an essential concept in the philosophy of artificial intelligence.

Alan Turing died in 1954 at the age of forty-one, after which his work became forefront identifying him as the founder of Computer Science[12].

Main achievements of Alan Turing:

- Designed the paper model of the world's first digitalcomputer
- Discoverer of Turing Machine, 1935
- Accepted as the inventor of Artificial Intelligence
- Awarded in 1945 for his wartime services
- He was one of the core team members who built The Bombe.
- Decoding the Enigma in 1942
- He was the key team member which decoded the 'Fish' cipher

The A. M. Turing Award is an annual prize given by the Association for Computing Machinery (ACM) to "an individual selected for contributions of a technical nature made to the computing community". It is stipulated that "The contributions should be of lasting and major technical importance to the computer field". The Turing Award is recognized as the highest distinction in Computer Science and considered as the Computer Science equivalent of a Nobel Prize. 2012 will always be remembered as the "Turing year" as ACM A.M. Turing Award Winners came together for the first time, to honor the 100th Anniversary of Alan Turing and reflect on his contributions, as well as on the past and future of computing.

As we look back on Alan Turning's contributions to the fields of Mathematics and Computer Science, it is hard not to think about his hypothetical reaction to our current lifestyle. Sure he may be thrilled with the computational capabilities of computers today, but he may as well be horrified with our personal dependence on mobile computing technologies. Was his love for artificial intelligence purely fantasy or are we beginning to push the ethical boundaries? It would have been magnificent for Turing to see his ideas become something tangible; we can only hope he would have embraced these advances.

References / Sources:

[1] http://www-groups.dcs.st-and.ac.uk/history/Biographies/Turing.html
[2] http://en.wikipedia.org/wiki/Entscheidungsproblem
[3] http://www.alanturing.net/turing_archive/archive/p/p01/p01.php
[4] http://www.fh-jena.de/~kleine/history/machines/Godfrey-VonNeumannComputerPlan.pdf
[5] http://www.ellsbury.com/enigmabombe.htm
[6] http://en.wikipedia.org/wiki/Turing_test
[7] http://www.loebner.net/Prizef/TuringArticle.html
[8] http://en.wikipedia.org/wiki/Alan_Turing
[9] http://www-groups.dcs.st-and.ac.uk/history/Biographies/Turing.html
[10] http://www.turing.org.uk/
[11] http://www.turing.org.uk/turing/scrapbook/computer.html
[12] http://www.turing.org.uk/sources/vonneumann.html
[13] http://plato.stanford.edu/entries/turing-machine/
[14] http://plato.stanford.edu/entries/computing-history/
[15] http://www.bbc.co.uk/news/technology-17662585
[16] http://plato.stanford.edu/entries/turing-machine/#5.2
[17] http://skepchick.org/2012/05/alan-turing-computing-pioneer-codebreaker-gay-icon/
[18] http://historysheroes.e2bn.org/hero/achievements/91/print
[19] http://skepchick.org/2012/05/alan-turing-computing-pioneer-codebreaker-gay-icon/
[20] http://en.wikipedia.org/wiki/Universal_Turing_machine
[21] http://www.scribd.com/doc/1532910/Von-Neumann-Computer-Architecture
[22] http://people.ee.ethz.ch/~ballisti/computer_topics/neumann.vs.turing.html
[23] http://turing100.acm.org/index.cfm?p=awardees
[24] http://en.wikipedia.org/wiki/Von_Neumann_architecture
[25] http://www.bbc.co.uk/news/technology-17662585

9. The Enigma and the Bombe - http://www.ellsbury.com/enigmabombe.htm
10. Turing test - http://en.wikipedia.org/wiki/Turing_test
11. COMPUTING MACHINERY AND INTELLIGENCE -http://www.loebner.net/Prizef/TuringArticle.html
12. Turing – the father of computer science - http://en.wikipedia.org/wiki/Alan_Turing

About the Author

Vivek Kulkarni is currently working as Principal Architect in Persistent Systems Ltd. He has a total experience of 18+ years in academia and software industry. He has served as a subject chairman for multiple subjects for the Board of Computer Engineering, University of Pune. He has written multiple books on "Theory of Computation".

He has also worked in organizations such as BMC Software, Symantec Corporation, and Tech-Mahindra. He is also one of the inventors for *System and Method of Universal Programming Language Conversion*, which has been internationally recognized and patented